27 c2
J. Como
0260

#7
Oncefaros
6-1-99

169.0976

PATENT APPLICATION

OIPE
NOV 0 6 1998
PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	:	Examiner: Not Yet Assigned
JAMES PHILIP ANDREW	)	
	:	Group Art Unit: 2721
Application No.: 09/161,770	)	
	:	
Filed: September 29, 1998	)	
	:	
For: A METHOD FOR DIGITAL DATA COMPRESSION	)	
	:	November 5, 1998

RECEIVED
NOV 0 9 1998
Group 2700

RECEIVED
MAY 3 1999
Group 2700

Assistant Commissioner for Patents
Washington, D.C. 20231

CLAIM TO PRIORITY

Sir:

Applicant hereby claims priority under the International Convention and all rights to which he is entitled under 35 U.S.C. § 119 based upon the following Australian Priority Applications:

PO 9510, filed September 29, 1997; and

PP 0776, filed December 8, 1997.

Certified copies of the priority documents are enclosed.

Applicant's undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should be directed to our new address given below.

Respectfully submitted,

Abigail Cousins

Attorney for Applicant

Registration No. 29,292

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200

RECEIVED
NOV 0 9 1998
Group 2700

A:\169-0976.CTP\rmd

RECEIVED
MAY 3 1999
Group 2700

Appln. No. 09/[illegible]
GAU 2721

AUSTRALIA

Patent Office
Canberra

I, KIM MARSHALL, MANAGER EXAMINATION SUPPORT AND SALES, hereby certify that the annexed is a true copy of the Provisional specification in connection with Application No. PP 0776 for a patent by CANON KABUSHIKI KAISHA and CANON INFORMATION SYSTEMS RESEARCH AUSTRALIA PTY LTD filed on 8 December 1997.

I further certify that the annexed specification is not, as yet, open to public inspection.

COMMONWEALTH OF AUSTRALIA
AUSTRALIA
PATENT OFFICE

WITNESS my hand this Twenty-ninth day of September 1998

KIM MARSHALL
<u>MANAGER EXAMINATION SUPPORT AND SALES</u>

S & F Ref: 403864

ORIGINAL

AUSTRALIA

Patents Act 1990

PROVISIONAL SPECIFICATION FOR THE INVENTION ENTITLED:

A Method of Data Compression

Name and Address of Applicant: Canon Kabushiki Kaisha, incorporated in Japan, of 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 146, JAPAN; Canon Information Systems Research Australia Pty Ltd, an Australian Company ACN No. 003 943 780, of 1 Thomas Holt Drive, North Ryde, New South Wales, 2113, AUSTRALIA

This invention is best described in the following statement:

A Method for Digital Data Compression

Field of the Invention

The present invention relates to the field of data compression with particular application to digital image compression. More particularly, the present invention discloses a digital image compression method using context entropy coding of discrete wavelet transform coefficients.

Background of the Invention

The field of digital data compression and in particular digital image compression has attracted great interest for some time.

In the field of digital image compression, many different techniques have been utilised. In particular, one popular technique is the JPEG standard which utilises the discrete cosine transform to transform standard size blocks of an image into corresponding cosine components. In this respect, the higher frequency cosine components are heavily quantised so as to assist in obtaining substantial compression factors. The heavy quantisation is an example of a "lossy" technique of image compression. The JPEG standard also provides for the subsequent lossless compression of the transformed coefficients.

Recently, the field of wavelet transforms has gained great attention as an alternative form of data compression. The wavelet transform has been found to be highly suitable in representing data having discontinuities such as sharp edges. Such discontinuities are often present in image data or the like. Whilst wavelet transforms can be applied to analogue signals, a discrete wavelet transform (DWT) is typically applied discrete or digital data.
Bit-plane decomposition and entropy coding DWT image bit plane techniques, based on a context of surrounding coefficients have demonstrated good results. However these techniques are performed in a breadth first manner. That is, for a predetermined region of a DWT image, or an entire DWT image, a bit-plane decomposition is performed and each

bit in a bit plane is entropy coded before each bit of another bit plane is coded. These techniques provide an embedded code, which is desirable for some applications, but require multiple passes through memory which contains the DWT image (coefficients) in the encoding or decoding process.

However in some applications multiple passes through memory are undesirable, particularly where memory access time is slow. For example, where images are stored on external memory, such as a CD-ROM, and a small amount of on-processor cache is available for executed of a coding or decoding process, access time to or from the external memory can limit the execution of the coding or decoding process.

Thus, there exists a need for techniques for bit-plane entropy (de)coding that provide a reduction in the number of memory accesses when compared with current techniques.

Although the preferred embodiments of the present invention will be described with reference to the compression of image data, it will be readily evident that the preferred embodiment is not limited thereto. For examples of the many different applications of Wavelet analysis to signals, reference is made to a survey article entitled "Wavelet Analysis" by Bruce et. al. appearing in IEEE Spectrum, October 1996 page 26 - 35.

Summary of the Invention

In accordance with the first aspect of the present invention there is disclosed a method of compressing data comprising the steps of:

a) applying a transform to the data to produce a plurality of transform coefficients, wherein each transform coefficient is expressible by a code representation comprising a plurality of symbols;

b) entropy encoding one of said symbols, not previously entropy coded, of a current transform

coefficient based on at least one other previously coded symbol;

c) repeating step b) a predetermined number of times for the current transform coefficient; and

d) processing another transform coefficient in accordance with steps b) and c).

Preferably said entropy encoding utilises coefficients surrounding a spatial location of a current transformed coefficient.

The method further preferably comprises quantising transformed coefficients to integer values including a sign bit and a predetermined number of coefficient bits. Ideally, the embodiments include wavelet transforming the data with each of the sub-band components of the wavelet transform being separately entropy encoded. The present invention is ideally suited to the compression of image data and can be implemented on a standard computer device. Other aspects of this invention are recited at the end of this specification.

Brief Description of the Drawings

Notwithstanding any other forms which may fall within the scope of the present invention, preferred forms of the invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figs. 1-3 illustrate the process of wavelet transforming image data;

Fig. 4 illustrates the steps involved in the encoder of the preferred embodiment;

Fig. 5 illustrates the steps in the decoder as constructed in accordance with the preferred embodiment;

Fig. 6 illustrates a data structure utilised by the preferred embodiment;

Fig. 7 illustrates a window for determining a context for a current coefficient;

Fig. 8 illustrates the process of utilising surrounding context for a current coefficient at bit-plane

n; and

Fig. 9 illustrates the process of block based coding in accordance with another embodiment of the present invention.

Description of Preferred and Other Embodiments

The preferred embodiment proceeds initially by means of a wavelet transform of image data. A description of the wavelet transform process is given in many standard texts and in particular the aforementioned book by Stollnitz et. al. An overview of the wavelet process will now be described with reference to the accompanying drawings.

Referring initially to Fig. 1, an original image 1 is transformed utilising a Discrete Wavelet Transform (DWT) into four subimages 3-6. The subimages or subbands are normally denoted LL1, HL1, LH1 and HH1. The one suffix on the subband names indicates level 1. The LL1 subband is a low pass decimated version of the original image.

The wavelet transform utilised can vary and can include, for example, Haar basis functions, Daubechies basis functions etc. The LL1 subband is then in turn utilised and a second Discrete Wavelet Transform is applied as shown in Fig. 2 giving subbands LL2 (8), HL2 (9), LH2 (10), HH2 (11). This process is continued for example as illustrated in Fig. 3 wherein the LL4 subband is referred to as an octave band filter bank with the LL4 subband being referred to as the DC subband. Obviously, further levels of decomposition can be provided depending on the size of the input image.

Each single level DWT can in turn be inverted to obtain the original image. Thus a J-level DWT can be inverted as a series of J-single level inverse DWT's.

To code an image hierarchically the DC subband is coded first. Then, the remaining subbands are coded in order of decreasing level. That is for a 4 level DWT, the subbands at level 4 are coded after the DC subband (LL4). That is the HL4, LH4 and HH4 subbands. The subbands at

level 3 (HL3, LH3, and HH3) are then coded, followed by those at level 2 (HL2, LH2 and HH2) and then level 1 (HL1, LH1 and HH1).

With standard images, the encoded subbands normally contain the "detail" information in an image. Hence, they often consist of a sparse array of values and substantial compression can be achieved by quantisation of the subbands and efficient encoding of their sparse matrix form.

In the preferred embodiment, an effective compression of the subbands is provided through the utilisation of relationship between adjacent coefficients in the DWT subband.

The encoding proceeds via a depth first approach. That is, each bit of a current coefficient is encoded, based on the context of bits, of previously coded surrounding coefficients, on a current bit-plane (bit-plane number n) and on whether or not a most significant bit (msb) of the current coefficient has been coded (ie. the msb number is greater than bit-plane n). By using a limited window a number contexts used in the encoding takes on a relatively small range of values, and hence there are a limited number of contexts. For efficient entropy coding such a small number of contexts is desired. After substantially all the bits of the current coefficient are encoded, a sign (positive "+" or negative "-") of the current coefficient, if the current coefficient is not zero, is transmitted or encoded as is. Optionally entropy coding of a sign of the coefficient could also be used at this stage.

An overview of the coding process is illustrated 20 in Fig. 4, while the decoding process is illustrated 30 in Fig. 5, in the form of structure diagrams. Typically structure diagrams are read with a left operation item being performed before a right operation item.

Turning initially to Fig. 4, a digital image is transformed 21 using a Discrete Wavelet Transform into

several subband components as previously described. Each subband is preferably coded in a hierarchical order. An encoding of the transform coefficients is performed 23 by a binary arithmetic coder. Each coefficient 24 in each subband 25 is iteratively processed by a quantisation 22 then an entropy coding by a binary arithmetic coder 23. As illustrated in Fig. 6, each coefficient in a subband is quantised to an integer value 26 having a predetermined number of bits L, and conceptually represented in a binary format with a sign bit s. Each bit 27 is preferably coded in order from a most significant bit-plane to a least significant bit-plane. A bit 27 of the integer value 26 is entropy coded using a context formed from a consideration of surrounding bits, and whether or not a msb of the integer value 26 has been entropy coded.

As illustrated in Fig. 5, at the decoder the operation of the encoder is reversed (in as much as this is possible with quantisation). A predetermined portion of an encoded bit-stream (ie. the result of the encoder) is decoded 33 by a binary arithmetic decoder to provide a quantised coefficient. The quantised coefficient is inverse quantised 31. This process is looped through (34,35) in an, iterative manner, to produce each coefficient of each subband. Finally an inverse Discrete Wavelet Transform is performed 32 on the resulting subbands to give the output image. This process need not be iterative. For example, the entire bit-stream can be decoded by the binary arithmetic decoder 33 to provide substantially all the quantised coefficients and then these coefficients can be inverse quantised 31 before applying the inverse Discrete Wavelet Transform 32.

As noted previously, in the encoding process the discrete wavelet transform coefficients are quantised 22 to integer values. Let c represent a coefficient value and d its quantised values. Then the quantisation is performed as,

$$d = fix\left(\frac{c}{q}\right)$$

where q is a predetermined quantisation factor and fix is defined by,

$$fix(x) = \begin{cases} \lfloor x \rfloor x \geq 0 \\ \lceil x \rceil x \leq 0 \end{cases}$$

were $\lfloor\ \rfloor$ is the round down to nearest integer operator and $\lceil\ \rceil$ is the round up to nearest integer operator. At the encoder each coefficient in a subband is quantised to an integer value using this equation.

The inverse quantisation is given by,

$$c = q \times d + sign(d) \times \frac{q}{2}$$

where,

$$sign(d) = \begin{cases} -1 d \leq 0 \\ 0\ \ d = 0 \\ 1\ \ d \geq 0 \end{cases}$$

At the decoder each coefficient is inverse quantised using this inverse quantisation equation. The quantisation factor q can vary from subband to subband, or it can be fixed for the whole image. It can be coded in the header of the compressed image.

<u>**Coefficient Coding and Decoding**</u>

As shown in Fig. 6, each quantised coefficient is an integer value represented in a binary format with a sign bit. For the purpose of the description of the preferred embodiment, it is assumed with 15 bits and an extra sign bit (ie. L=16)). Thus

$$d = sign(d) \times b_{14} b_{13} \cdots b_0$$

where b_n is binary bit n.

The coefficient d is coded by entropy coding the bits b_{14}, .. b_0 in order, and a sign bit for non-zero coefficients. Bit b_n of a coefficient is coded with a binary entropy coder (eg. binary arithmetic coder) based on the context

determined by a bit pattern formed from: bit n of each of the surrounding coefficients; whether the most significant bit of a current coefficient has been coded (ie whether the msb number is greater than n); and on whether the msb of any of the surrounding coefficients have been coded. A description of surrounding coefficients and context thus formed is given below. A sign of each coefficient not quantised to zero can be coded as is or entropy coded based on surrounding coefficient signs.

At the decoder the quantised coefficient d is reconstructed by simply entropy (eg. binary arithmetic) decoding bits b_{14}, .. b_0 and a sign bit for non-zero coefficients.

Surrounding Coefficient Context

Turning to Fig. 7, a subband eg. 35 is coded in raster scan order from top to bottom and left to right. If a current coefficient to be coded is marked with a cross 36, the surrounding coefficients are considered to be the four surrounding coefficients indicated by the four empty squares 37-40. Preferably the surrounding coefficients are selected by a window with a shape as indicated in Fig. 7. If the cross in the window is aligned with the current coefficient the surrounding coefficients 37-40 are defined to be the coefficients that fall within the window.

The window illustrated follows a raster scan order. Hence, when the current coefficient is being decoded the surrounding coefficients have already been decoded, and thus the decoder knows whether or not they are non zero.

Referring to Fig. 8, there is shown a single bit-plane 50, bit-plane n, of the DWT coefficients. The window, described with reference to Fig 7., includes four bits at bit-plane n labelled $c_2, \ldots, c_5$ of surrounding coefficients 42-45. Preferably additional flags other than the surrounding bits $c_2, \ldots, c_5$ are used to determine a context.

In the preferred embodiment additional flags include a flag, c_0, that indicates whether or not a current coefficient 41 has a msb that has been previously entropy coded, and/or a flag, c_1, indicating whether or not any the surrounding coefficients have a msb which has been previously entropy coded. That is, if encoding the coefficient is performed from a highest value bit-plane to a lowest value bit-plane then c_0 represent whether or not there is a set (1) bit of a current coefficient in bit-planes above the current bit-plane, and c_1 represent whether or not there is a set (1) bit on any of the surrounding coefficient 42-45 in bit-planes above or in the current bit-plane

A context for bit n of the current coefficient 41 is determined, in the preferred embodiment, by 6 bit binary number $c_0, c_1, \ldots, c_5$. Bit c_0 of the context is set (ie has value 1) if the most significant bit (msb) of the current coefficient has already been coded. That is the msb of the current coefficient is in bit plane n+1 or greater. Bit c_1 is set if any one of the four surrounding coefficients 42-45 has a msb in bit plane n or greater. Finally bits c_2, c_3, c_4, and c_5 are determined by a bit pattern of the four surrounding coefficients 42-45. In this case, for example, there are $2^6 = 64$ contexts, since there are 64 different permutations of binary bits $c_0, c_1, \ldots, c_5$.

The context formation described above is causal in that the context can be formed from previously coded information. In this manner the decoder can form the same context as the encoder. For the window in Fig. 8, following the raster scan order 46, when bit n of the current coefficient is being decoded the surrounding coefficients 42-45 have already been decoded, and thus the decoder knows bit n for each of these coefficients, 42-45, and whether or

not the msb of these coefficients is in a bit-plane greater than or equal to bit-plane *n*. The decoder also knows whether the msb of the current coefficient is in a bit-plane greater than or equal to bit-plane *n*+1. Thus the decoder can form the same context as used at the encoder.

Optionally different windows and different information about previously coded bits in coefficients can be used to form a variety of different sets of contexts without departing from the scope and spirit of the invention. For example another window configuration can be used to determine a context. Typically information used to determine a context is causal. That is, surrounding coefficients (or bits) used in the determination of the context are to be processed before a current coefficient. Preferably in a raster scan (or coding) order. A small local window is also desired because it minimises the local memory buffering requirements. In addition, a small number of contexts is preferred to minimise the amount of memory required for the entropy coding, and to prevent context dilution. The four coefficient window, and whether or not the msb has been coded, is used, herein, as a good compromise between complexity, which grows with the number of contexts, and compression efficiency, which increases with the number of contexts, at least up to a predetermined number of such contexts.

Typically, once the most significant bit of a coefficient has been coded, a different context entropy coder could be used: one that is based more on the current coefficient to be coded than on the surrounding coefficients. Optionally, it is possible to simple code these bits without entropy coding.

Context Entropy Coding

As noted previously, each bit of a current coefficient is coded with a context based entropy coder. Preferably, this is a standard arithmetic coder. Arithmetic coding is

described in Witten et. al., "Arithmetic coding for data compression", Communications of the ACM, Volume 30, No. 6, June 1987.

Typically Arithmetic coding relies on assigning a probability of occurrence of a symbol, in a plurality of symbols to be encoded. Hence one option of the preferred embodiment is to assign a predetermined probability to each of the 64 different contexts, preferably so that each context has a probability value indicative of a likelihood that a current symbol has a resulting value. The contexts and their corresponding probabilities form a look up table that is duplicated for a decoder, so the decoder can mimic the encoder. Naturally, a fixed probability approach works well on stationary images but in reality not all images are stationary and while a performance of the encoder/decoder is adequate, using a fixed probability approach, both for stationary and non-stationary sources an adaptive arithmetic coder is preferred. For non-stationary sources (images) an adaptive arithmetic (de)coder will overcome changes in the probability distribution of each symbol.

Another embodiment of the present invention can be described as a variation on the preferred embodiment in which a combination of a breadth first and a depth first approach is adopted. In the present embodiment an image is, transformed, quantised and divided into a plurality of blocks. Preferably each block comprising a fixed number of quantised transformed coefficients. A predetermined block of transformed coefficients is processed in a breadth first manner. That is, each coefficient in the block and each symbol of each coefficient is processed in a sequential manner. For example a first symbol of a first coefficient is encoded substantially as described in the preferred embodiment, then a first symbol of a second coefficient is encoded, then a first symbol of a third coefficient is encoded etc. Until all first symbols of the coefficients in the block have been encoded. Next all the second symbols

of each coefficient in the block are encoded. This is continued across (breadth first) each coefficient in the block until substantially all coefficients and their corresponding symbols have been encoded. Another block of coefficients is then processed in a substantially similar manner until substantially all the blocks and hence substantially all coefficients of the transformed image are encoded.

Essentially, in the present embodiment the depth first approach upon each transform coefficient described, above, with reference to the preferred embodiment is adopted on a block by block basis. However, the coefficients within each block are encoded in a breadth first approach.

Referring to Fig. 9, there is shown an array 60 of transform coefficients 61 , representing a set of quantised transform coefficients 61 of an image. The array 60 of coefficients 61 are divided into a plurality of blocks and an example of one such block 62 is also shown in Fig. 9. The block 62 is a 2x2 block comprising four coefficients 63-66 which are typically encoded on a symbol by symbol basis in a zigzag fashion 67. That is, a first symbol of the first coefficient 63 of the block 62 is encoded based on a context determined for the first symbol substantially as described in the preferred embodiment. Next a first symbol of the second coefficient 64 of the block 62 is encoded. Followed by an encoding of a first symbol of the third coefficient 65 of the block 62 and similarly a first symbol of the fourth coefficient 66 is encoded. Next a second symbol of all the coefficients 63-66, of the block 62, are encoded in substantially the same manner. This is repeated until substantially all symbols of the coefficient of the block 62 are encoded before another block is encoded.

A determination of an ideal block size (ie. the number of coefficients per block) may vary from application to application and/or an amount of available high speed cache

memory, however a preferred block size is one than exploits both advantages of a depth first approach and a breadth first approach. Advantages of the breadth first approach include having knowledge of previously encoded symbols of an entire block. For example, as noted previously, once all the first symbols of each coefficients of a block is encoded a context of second symbols of each coefficient can benefit from a knowledge of the first symbols encoded across the entire block. Advantages of a depth first approach include on each block is that an array of coefficients can be read on a block by block basis rather than a symbol by symbol basis required in a purely breadth first approach, thus reduce the number of accesses to the array.

Typically, the array 60 of coefficients is stored in memory such as RAM or on external memory such as a hard disk drive for example. Access time for such memory read and/or writes is typically long when compared to on processor cache memory. Consequently, increased performance can be had by caching the block 62 in local cache (high speed)memory rather than access coefficients from the array 60 stored in a hard disk drive or lower performance memory. Images are generally large with respect to the number of bytes per file when compared to, for instance, a text file and hence it is not unreasonable to store a library of images on a hard disk drive. The method of the present embodiment therefore reduces the number of accesses to the array 60 stored in external memory than the number of accesses of a purely breadth first approach. Thus improving performance without the need to cache substantially the entire array 60.

A context of a current symbol in the block based coding technique of the present embodiment is determined substantially as hereinbefore described with reference to the preferred embodiment. Thus in addition to caching the block 62, surrounding coefficients 68 are also cached.

Naturally, the number of coefficients 68 and which coefficients are to be cached, in addition to the block 62, depends upon a choice of window and/or flag bits for determining a context.

The embodiments also have application to other image formats. For example, full colour images can be encoded via separate colour channels or the usual chrominance compression techniques as utilised in the JPEG standard can be applied so as to produce reduced chrominance data. Further, the preferred embodiment is described with reference to binary symbols (ie 0 and 1) and bit-planes, however the embodiment, of the depth first approach, can also be implemented with an n-ary representation of coefficients. That is, each coefficient can be represented by a plurality of different symbols (ie. n symbols for an n-ary representation). Still further, not all symbols of a current coefficient need to be encoded before encoding a next coefficient. For example, in an eight bit representation of coefficients, four bits of each coefficient can be encoded on each pass (raster scan). Hence in two passes of memory each coefficient will be encoded.

Additionally, the principles of the embodiments can be equally extended to other forms of data such as sound data etc. and the preferred embodiment has application wherever wavelet transforms are suitable. Additionally, the embodiments can be applied to other forms of transformed data for example, the discrete cosine transform process in addition to the wavelet packet and cosine packet transform techniques as described in the aforementioned survey article.

The embodiments are ideally implemented on a suitably programmed computer device.

It would be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present invention as shown in the specific

embodiments without departing from the spirit or scope of the invention as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive.

Aspects of the invention

The following numbered paragraphs recite aspects of the present invention:

1. A method of compressing data comprising the steps of:

a) applying a transform to the data to produce a plurality of transform coefficients, wherein each transform coefficient is expressible by a code representation comprising a plurality of symbols;

b) entropy encoding one of said symbols, not previously entropy coded, of a current transform coefficient based on a context of surrounding symbols;

c) repeating step b) a predetermined number of times for the current transform coefficient; and

d) processing another transform coefficient in accordance with steps b) and c).

2. The method recited in paragraph 1, wherein said context of surrounding symbols is determined from previously encoded coefficients.

3. The method recited in paragraph 1, wherein the method includes a further step of quantising said transform coefficients.

4. The method recited in paragraph 1, wherein said predetermined number of times is consistent with an encoding of substantially all of the symbols of the current transform coefficients.

5. The method recited in paragraph 1, wherein said context is determined from an arrangement of surrounding symbols.

6. The method recited in paragraph 5, wherein said surrounding symbols are previously encoded symbols.

7. The method recited in paragraph 5, wherein said context includes a first flag which indicates whether or not a most significant symbol of the current transform coefficient has been encoded.

8. The method recited in paragraph 7, wherein said context includes a second flag which indicates whether or not a most significant symbol, of at least one transform coefficient spatially adjacent to the current transform coefficient, has been encoded.

9. A method of compressing data comprising the steps of:

a) applying a transform to the data to produce a plurality of transform coefficients, wherein each transform coefficient is expressible by a binary code representation having a plurality of bits;

b) entropy encoding one of said bits, not previously entropy coded, of a current transform coefficient based on a context of surrounding bits;

c) repeating step b) a predetermined number of times the current transform coefficient; and

d) processing another transform coefficient in accordance with steps b) and c).

10. The method recited in paragraph 9, wherein said context of surrounding bits is determined from previously encoded coefficients.

11. The method recited in paragraph 9, wherein the method includes a further step of quantising said transform coefficients.

12. The method recited in paragraph 9, wherein said context of surrounding bits includes information as to whether or not a most significant bit of the current transform coefficient has been encoded.

13. The method recited in paragraph 9 or 12, wherein said context of surrounding bits includes information as to whether or not a most significant bit of at least one transform coefficient spatially adjacent, to the current transform coefficient, has been encoded.

14. The method recited in paragraph 9, wherein said transform coefficients are represented in a bit-plane representation and said surrounding bits are bits in a

current bit-plane.

15. The method recited in any one of the preceding paragraphs, wherein said entropy coding is performed by an arithmetic coder.

16. The method recited in any one of the preceding paragraphs, wherein said transform is a Discrete Wavelet Transform.

17. An apparatus when implementing the method as set out in any one of the preceding paragraphs.

DATED this Eighth Day of December 1997

Canon Kabushiki Kaisha

Canon Information Systems Research Australia Pty Ltd

Patent Attorneys for the Applicants

SPRUSON & FERGUSON

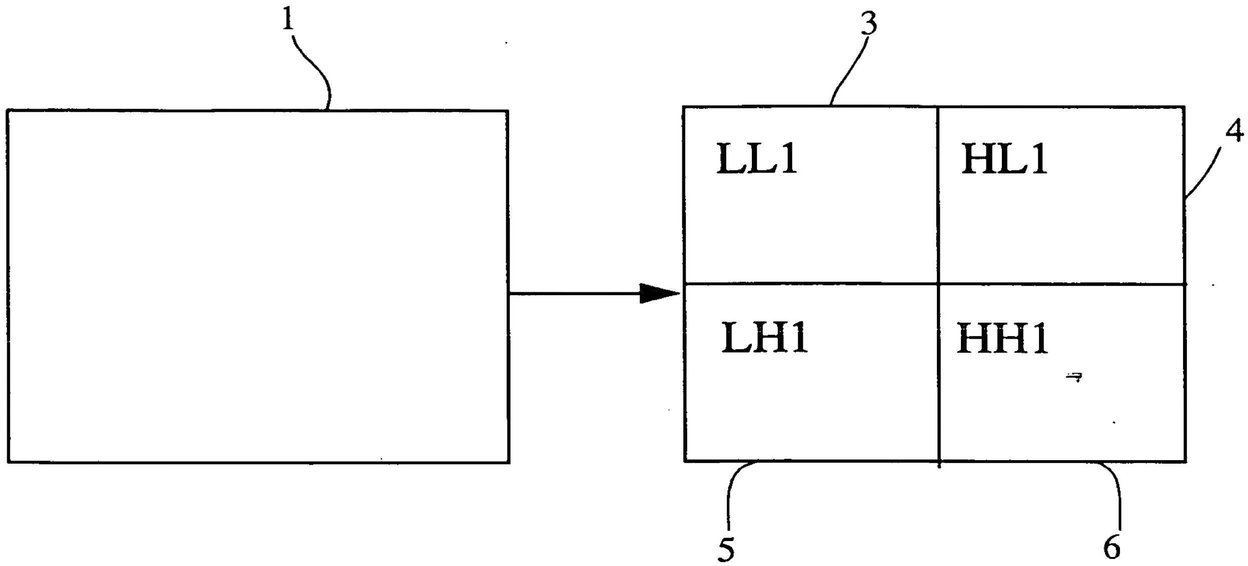


Fig 1

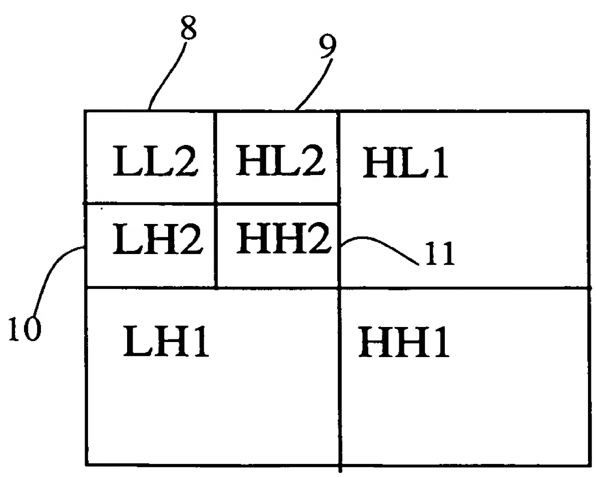


Fig 2

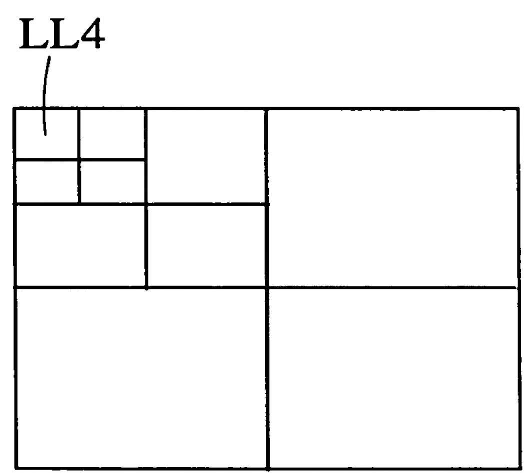


Fig 3

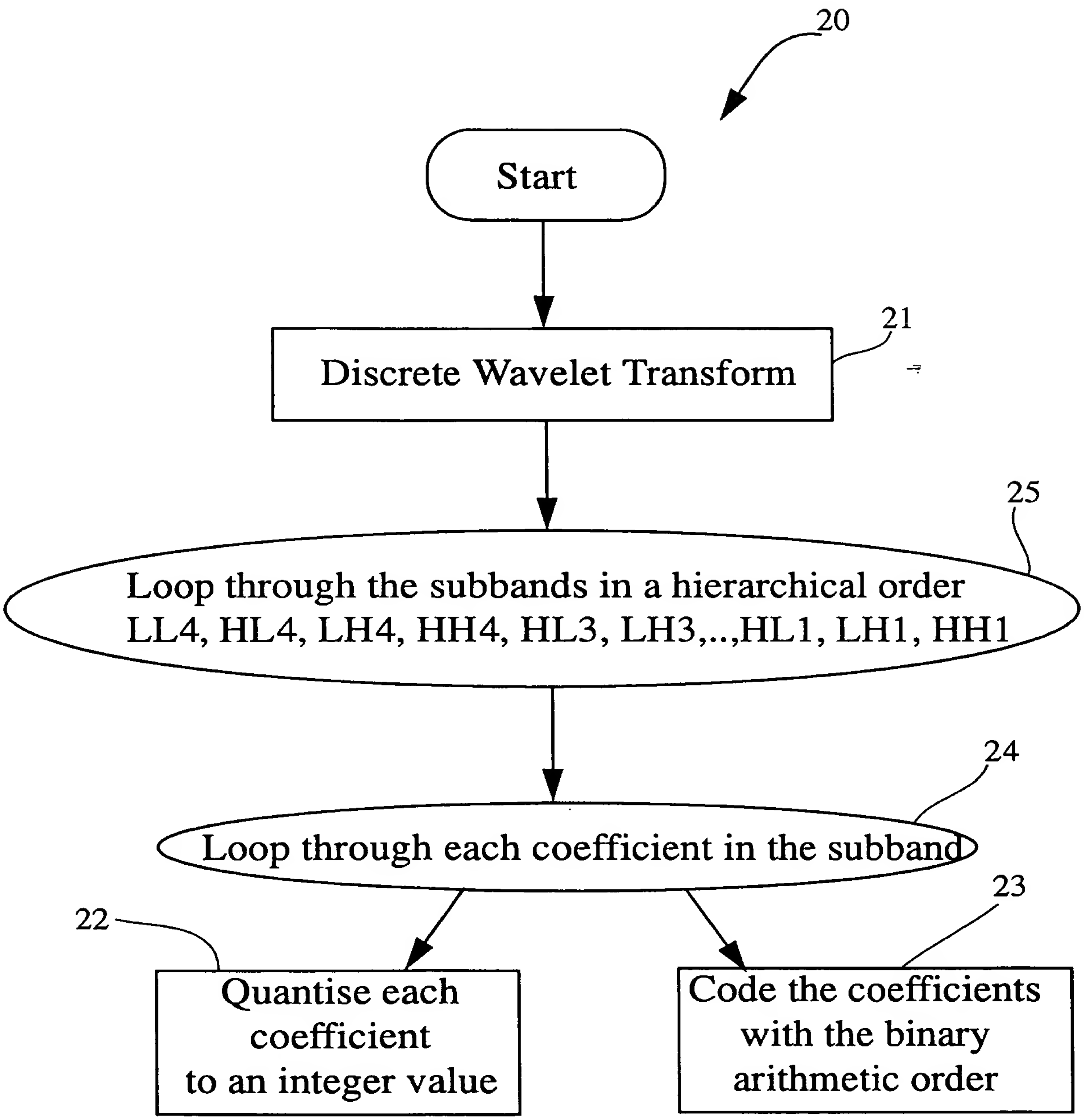

20
Start
Discrete Wavelet Transform
21
25
Loop through the subbands in a hierarchical order
LL4, HL4, LH4, HH4, HL3, LH3,..,HL1, LH1, HH1
24
Loop through each coefficient in the subband
23
22
Quantise each coefficient to an integer value
Code the coefficients with the binary arithmetic order

Fig 4

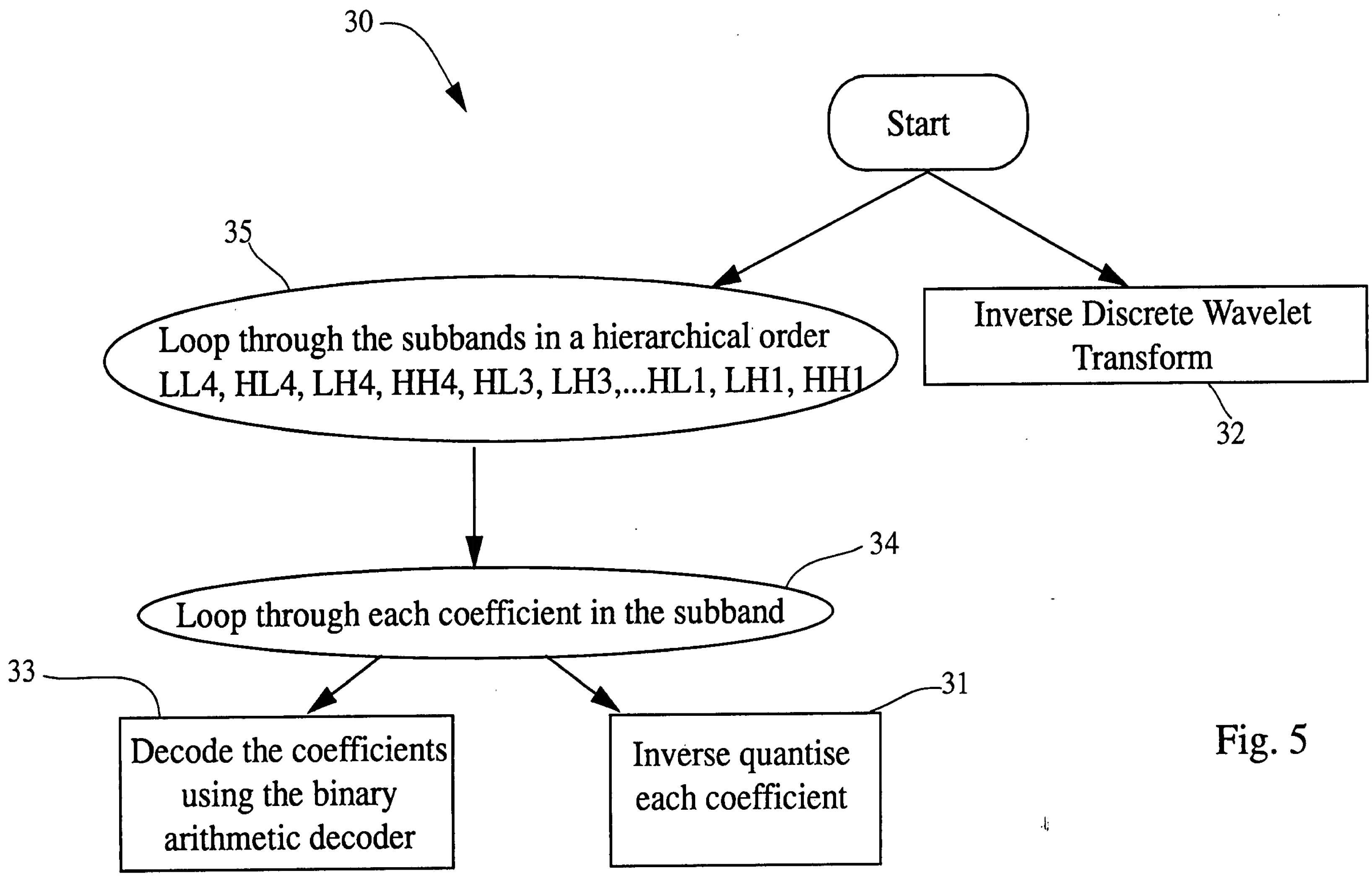

30
Start
35
Loop through the subbands in a hierarchical order LL4, HL4, LH4, HH4, HL3, LH3,...HL1, LH1, HH1
Inverse Discrete Wavelet Transform
32
34
Loop through each coefficient in the subband
33
Decode the coefficients using the binary arithmetic decoder
31
Inverse quantise each coefficient

Fig. 5

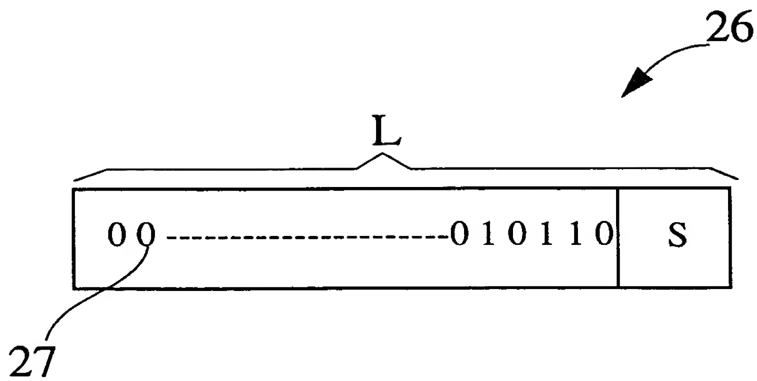


Fig 6

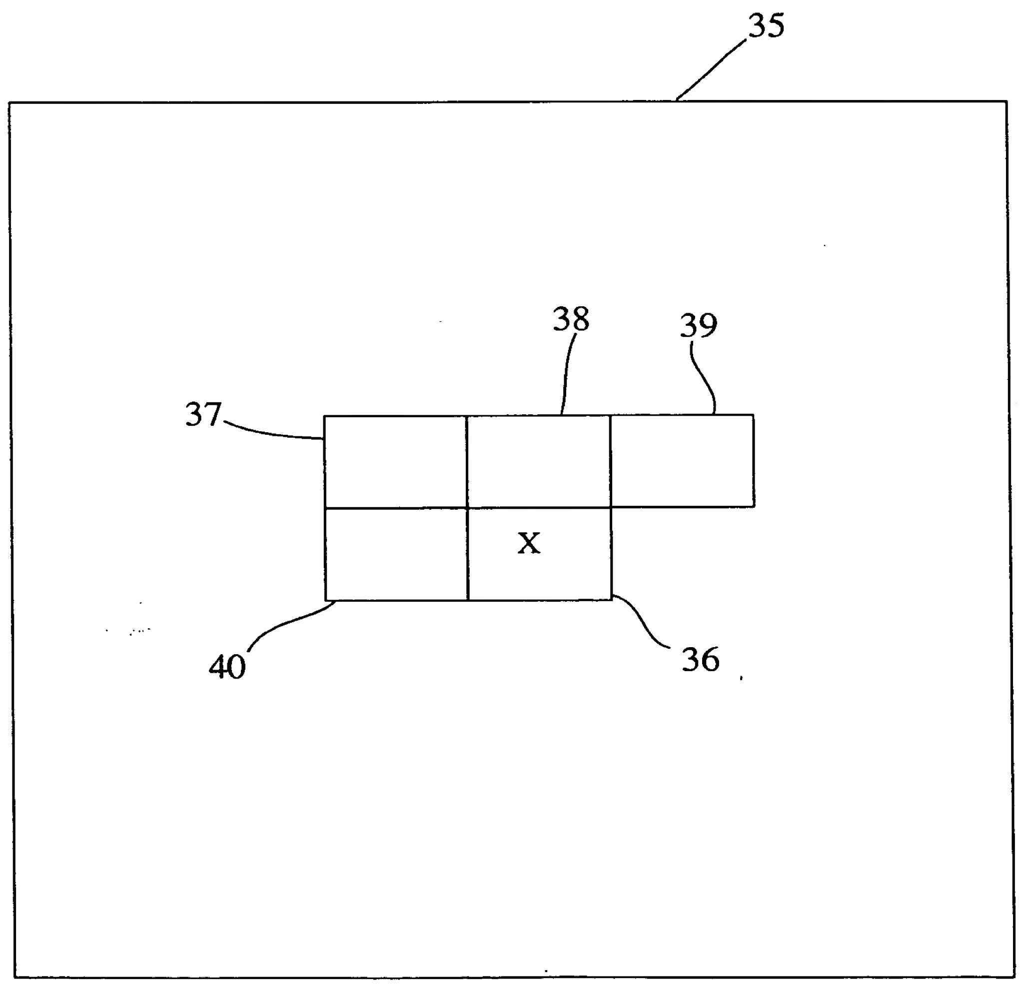
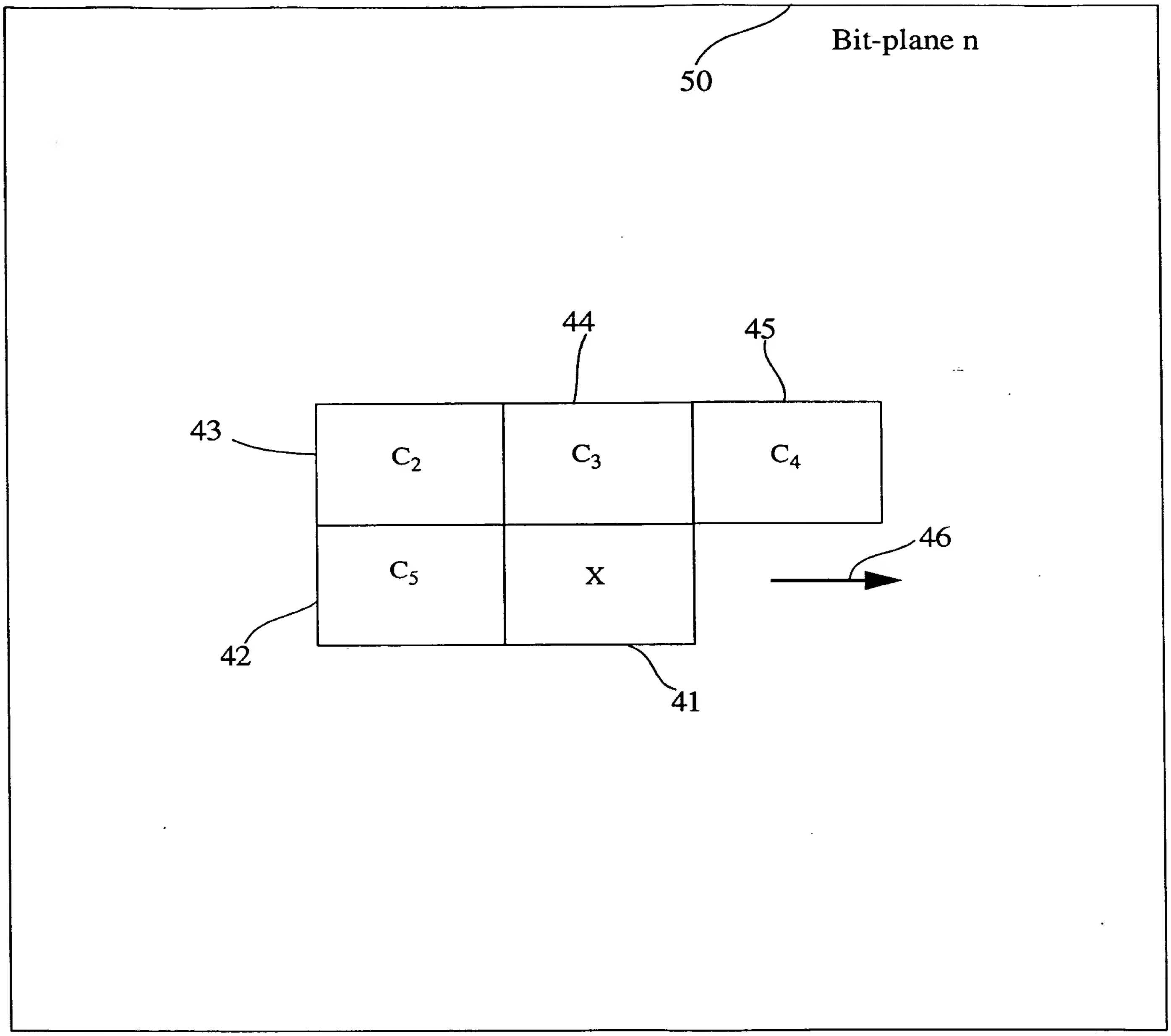


Fig. 7



$$C_0 = \begin{cases} 1 & \text{if a most significant bit of current coeffieient has been encoded} \\ 0 & \text{if a most significant bit of current coefficient has not been encoded} \end{cases}$$

$$C_1 = \begin{cases} 1 & \text{if any one or more of surrounding coefficient's most significant bits have been coded} \\ 0 & \text{if none of surrounding coefficient's most significant bits have not been coded} \end{cases}$$

Fig. 8

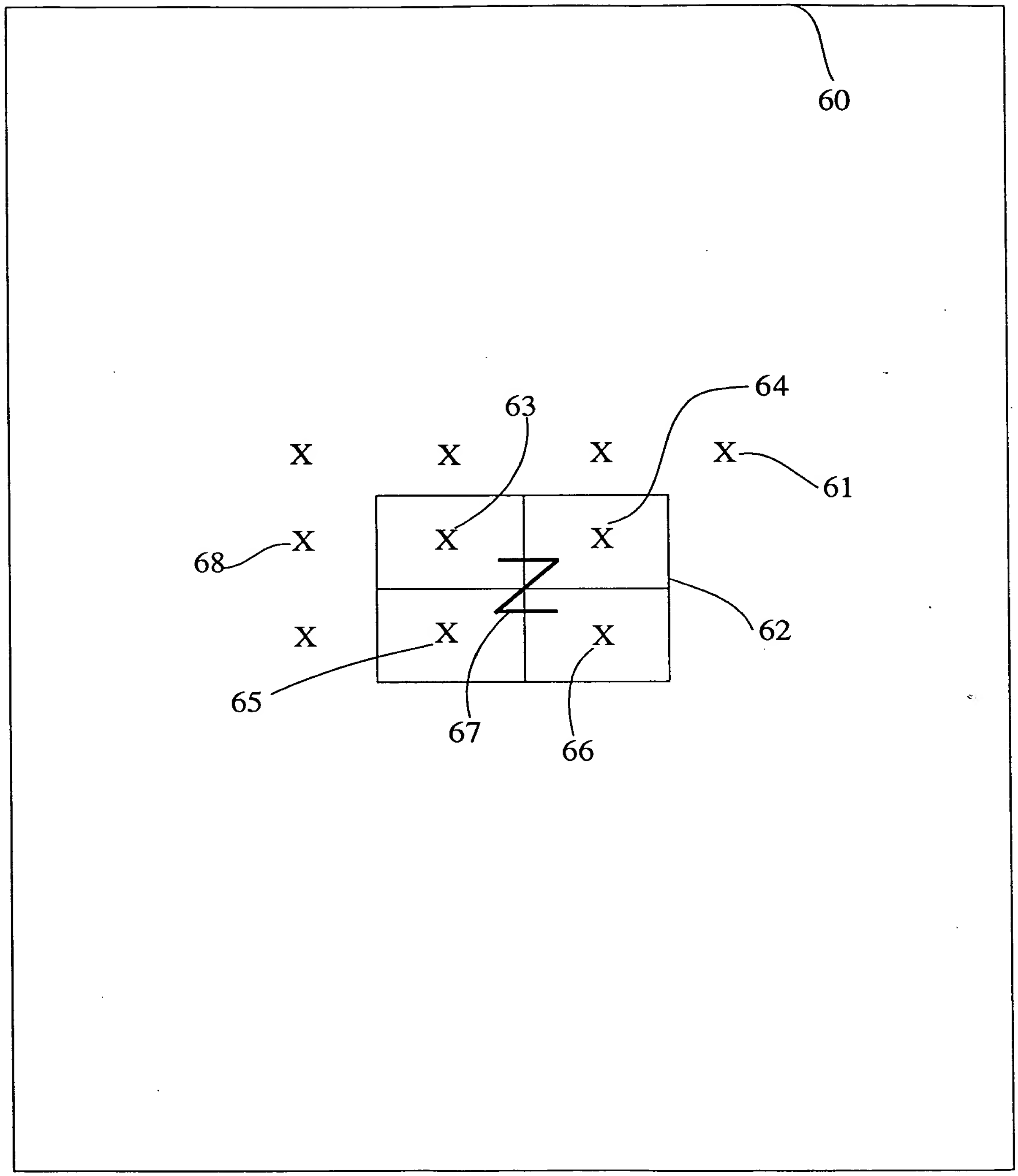


Fig. 9